

USP ICMC

# Sistema Digitais

Interface gráfica  
feito em C (linguagem de programação)

Ewerton P.S. Amaral  
2017

**Integrantes:**

**Nome:** Ewerton Patrick Silva do Amaral **numero Usp:**10346975

### Resumo do trabalho:

Foi feita interface gráfica para gerar imagem de uma simulação da máquina de estado finita, a interface gráfica notifica o estado, as saídas e as entradas. Além disso, a interface gráfica gera simulações da máquina de estado tanto em Melle quanto em Moore.

Para simular a máquina de estado foi utilizado o conceito de grafos desenvolvidos em trabalhos práticos da matéria de Introdução a Ciência da Computação 2 da USP. Para a construção da interface gráfica foi utilizado conceitos aprendidos em trabalhos práticos da matéria de Introdução a ciência da Computação 1 da USP, esse trabalho envolve manipulação e geração de imagem no formato .pgm.

### Informações sobre a imagem :

Cada estado é representado por uma bola na imagem FSM.pgm. De cima para baixo cada bola está relacionada diretamente a um valor da tabela verdade, indo de 1 até 9, cada uma dessas bolas é representada por uma letra alfabética. É demonstrado a seguir como se encontram os estados e as letras alfabéticas, esta imagem foi gerada durante a fase de criação do programa apenas para visualização dos estados:

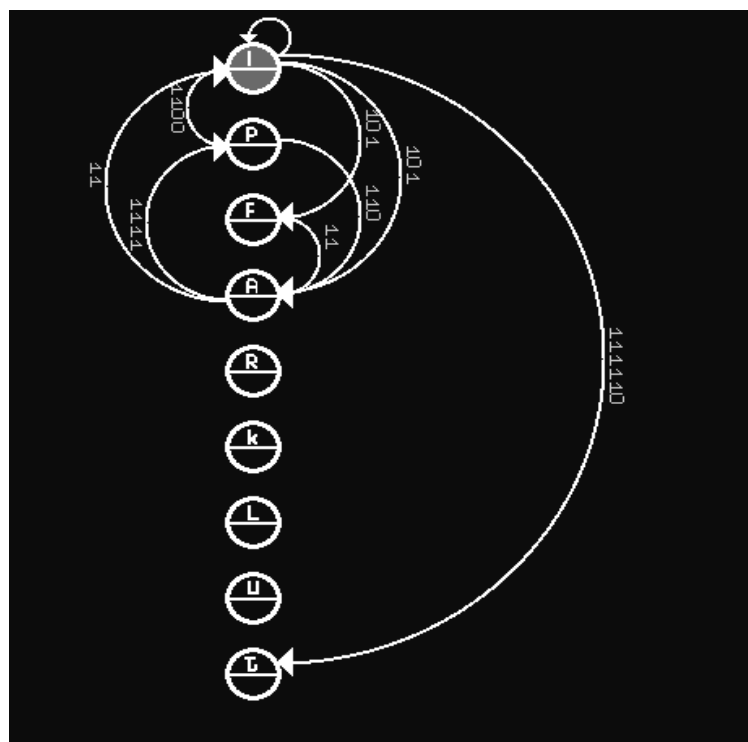


Imagem demonstrando onde se encontra as entradas e saídas em melle, sendo que as saidas e entradas são separadas por uma barra:

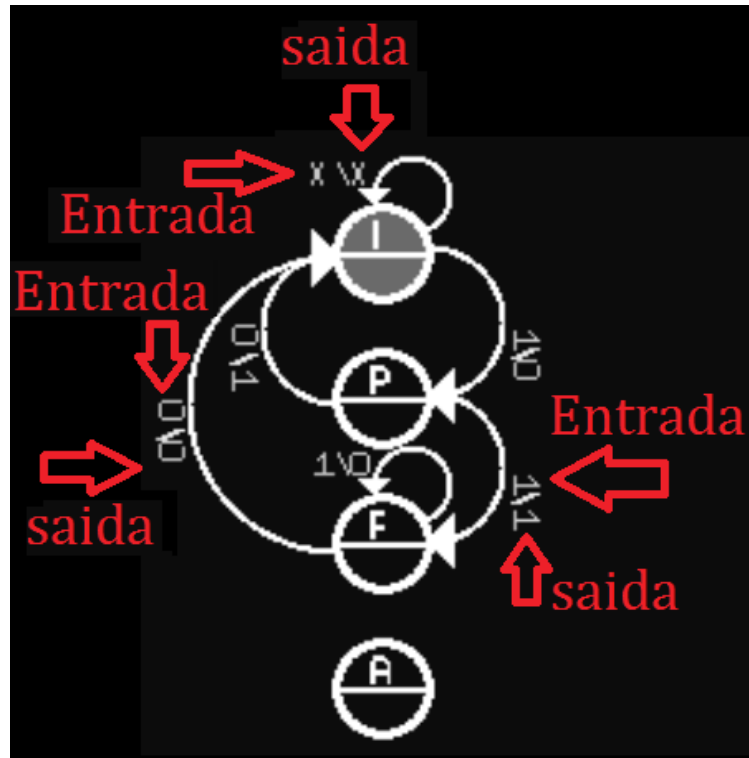
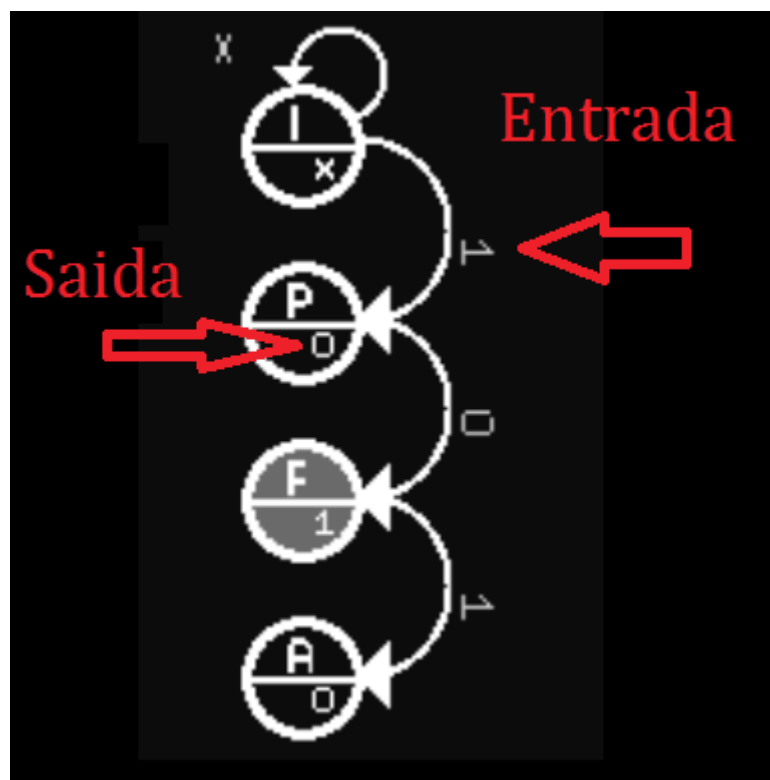


Imagem demonstrando onde se encontra as entradas e saídas em moore, sendo que as saidas e entradas são separadas por uma barra:

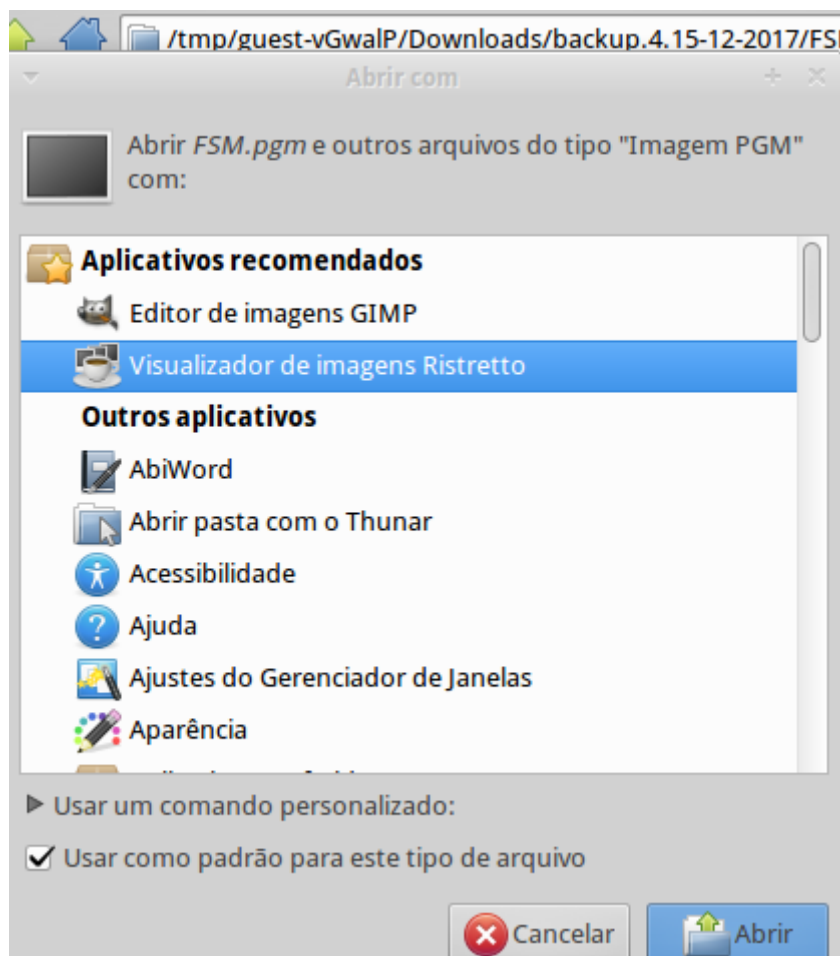


### Orientações para execução do programa:

Na pasta FSM se encontra o executável nomeado como “exe”, para sua execução é necessário abrir um terminal no linux e realizar a execução a partir do comando “./exe”

Apos executado deverá ser digitado: Melle(0) ou Moore(1), a quantidade de estado, quantidade de bit de entrada, quantidade de bit de saída e a tabela verdade. Na tabela verdade deve ser respeitado os espaçamentos entre cada informação digitalizada. Os “X” da tabela verdade deve ser escrito em maiúsculo, caso contrário não será reconhecido.

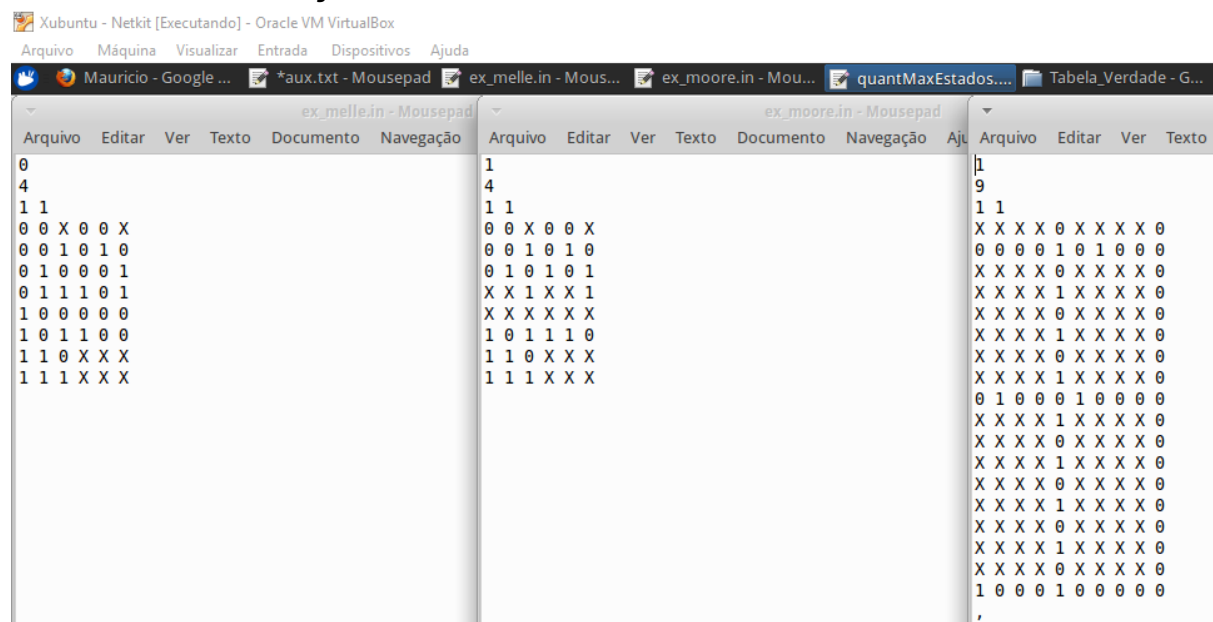
Após a digitalização correta da tabela verdade é notificado no terminal “Digite a condição para passar para o próximo estado”, isto indica que foi gerado uma imagem na pasta FSM, é necessário ir nesta pasta e abrir a imagem FSM.pgm no formato que se encontra na imagem a seguir ou no visualizador padrão do linux:



Com a imagem aberta poderá ser digitado as entradas diretamente no terminal, a máquina de estado sempre começa no estado 1 que é representado pela letra I. O terminal informará depois de digitado a entrada em qual estado você se encontrava e se houve alteração de estado. Os estado são preenchidos em cinza para indicar em qual estado se encontra, se for digitado corretamente a entrada para outro estado é atualizado a imagem juntamente ao terminal. Caso queira finalizar o programa após a execução basta digitar a virgula(“,”) no terminal que será finalizado todo o processo.

### **Exemplos de entradas prontas**

Na pasta FSM se encontra 3 arquivos demonstrando o formato correto da entrada, são eles: ex\_melle.in, ex\_moore.in e quantMaxEstados.in. A seguir é demonstrado esses arquivos e a forma correta da construção da tabela verdade .



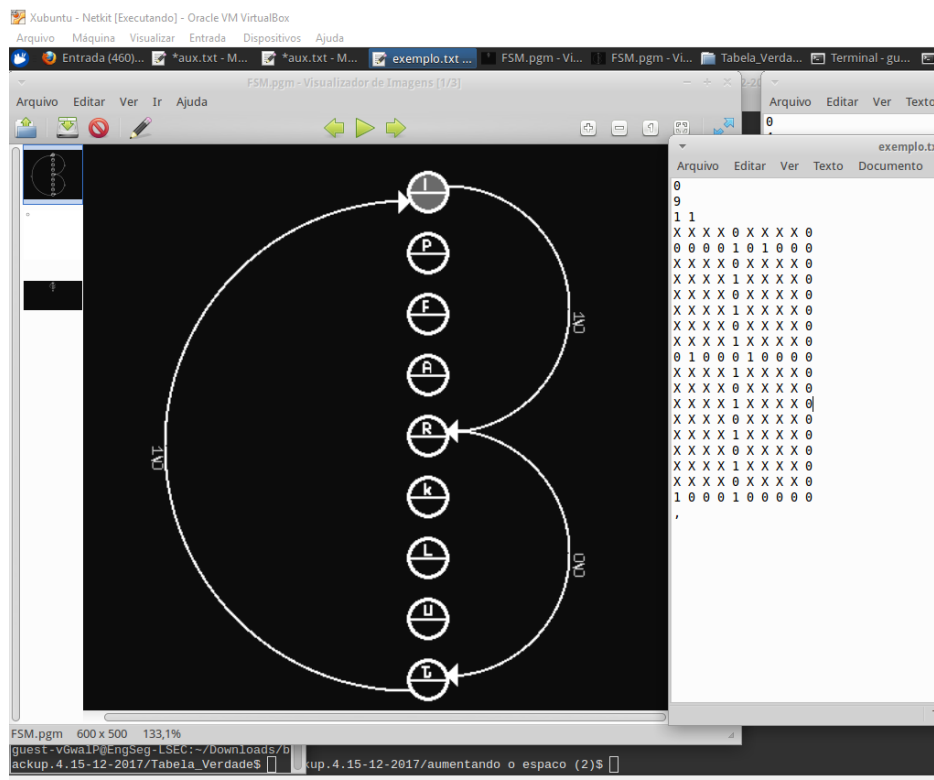
Eles podem ser usados para incluir nas entradas da execução do programa, contudo é necessário seguir os seguintes passos: Abra ele no linux, selecione todo o conteúdo e cope(ctrl + c), em seguida execute o programa no terminal pelo comando “./exe”, cole no terminal(ctrl+alt+v) e de entender, em seguida basta abrir a imagem FSM.pgm e digitalizar as entradas no terminal conforme descrito anteriormente nas orientações do programa.

### Tabela verdade

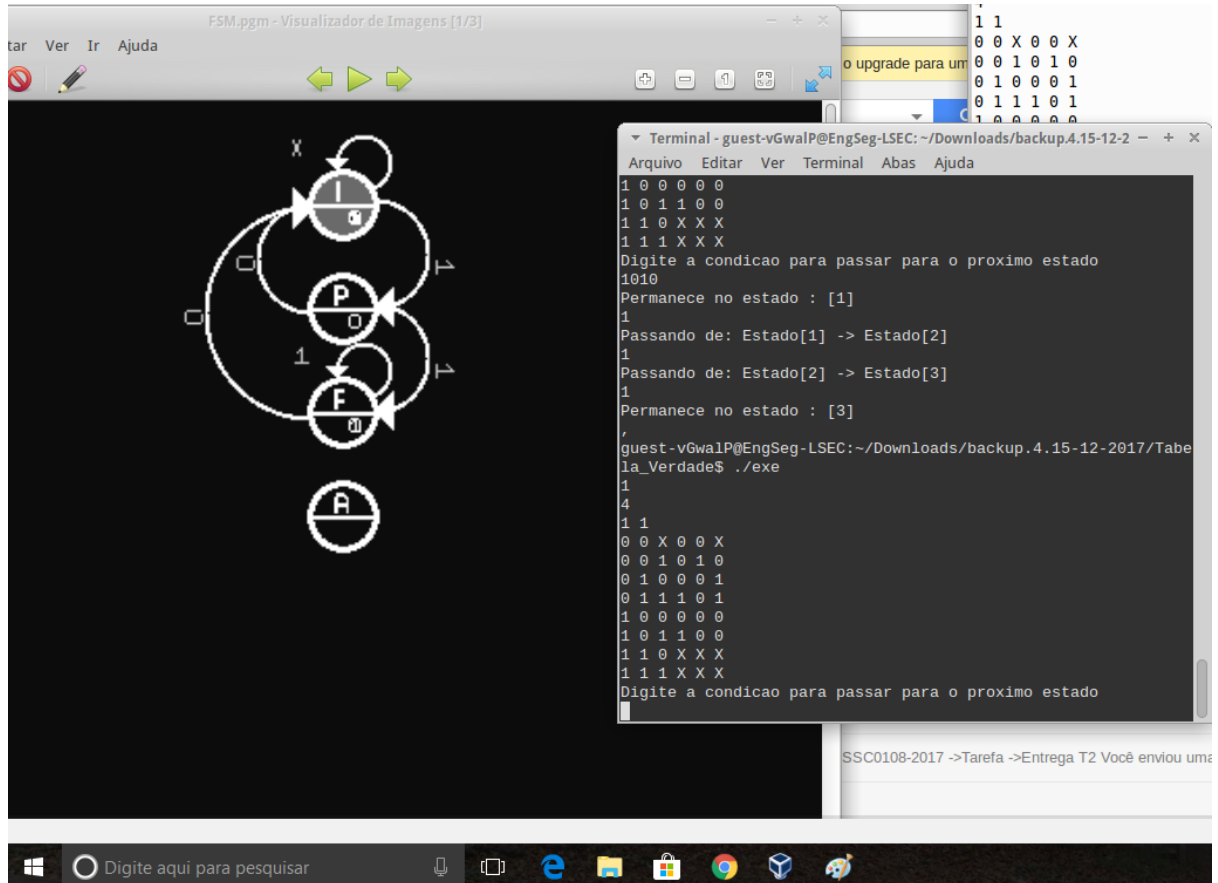
Vale lembrar que a tabela verdade varia de acordo com a quantidade de bits de entrada, saída e quantidade de estados, exemplo: digitando 9 para maquinas de estados, 1 bit de entrada e 1 bit de saída teremos 9 ( quantidade de estado) vezes 2 (quantidade máxima de possibilidades a partir do bit de entrada ou saída), sendo portanto necessário digitar 18 linhas de tabela verdade.

## Limitações do Programa

- Esse programa contém apenas 9 estados para representar as máquinas de estados



- Se for optado pela representação em moore e houver mais de uma saída para um único estado haverá uma sobreposição de imagem dificultando a identificação da saída



- Se houver muitas transições a partir de um estado na representação em melle as linhas podem acabar se sobrepondo as entradas e saídas representadas a cima de outras linhas.

## Imagens da execução do programa

EM MELLE:

The screenshot shows a computer screen with a file manager window titled "FSM.pgm - Visualizador de Imagens [1/3]" displaying a state transition diagram. The diagram has five states: A (bottom), F (middle), P (middle), I (top), and R (top). Transitions are labeled with inputs and outputs: A to F (1/0), F to P (1/0), P to I (1/0), I to R (1/0), R to I (1/0), I to P (0/1), P to F (0/1), and F to A (0/1). There is a self-loop on state I labeled X/X. A terminal window titled "Terminal - guest-vGwalP@EngSeg-LSEC: ~/Downloads/backup.4.15-12-2" is open, showing the execution of a program. The terminal output includes state transitions and a table of results.

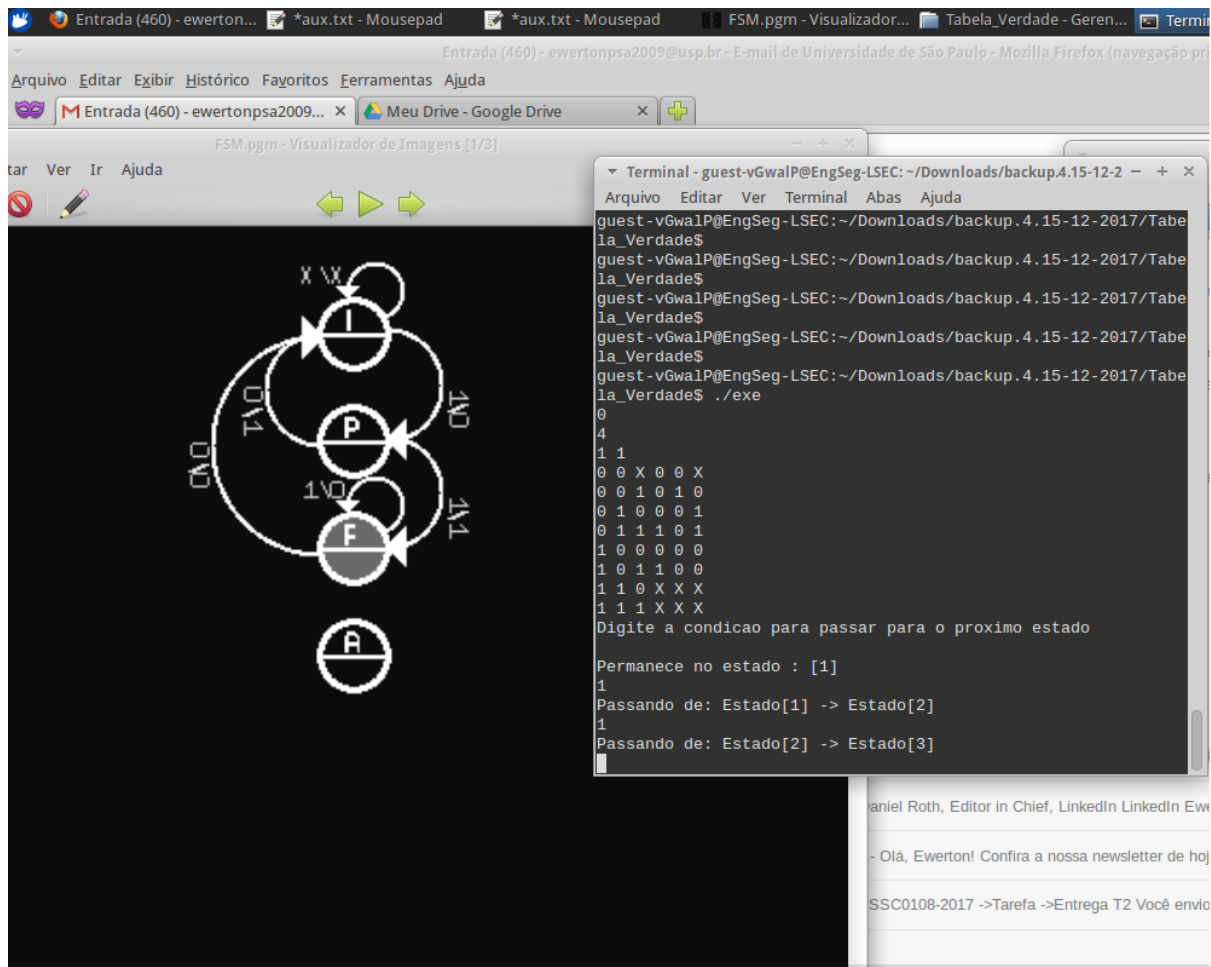
```
1
Passando de: Estado[2] -> Estado[3]
1
Passando de: Estado[3] -> Estado[3]
0
Passando de: Estado[3] -> Estado[1]
10110
Permaneço no estado : [1]

guest-vGwalP@EngSeg-LSEC:~/Downloads/backup.4.15-12-2017/Tabe
la_Verdade$ vim bibliotecaT2.c
guest-vGwalP@EngSeg-LSEC:~/Downloads/backup.4.15-12-2017/Tabe
gcc -o exe t2.c bibliotecaT2.c readLine.c -lm -I.
guest-vGwalP@EngSeg-LSEC:~/Downloads/backup.4.15-12-2017/Tabe
0
4
1 1
0 0 X 0 0 X
0 0 1 0 1 0
0 1 0 0 0 1
0 1 1 1 0 1
1 0 0 0 0 0
1 0 1 1 0 0
1 1 0 X X X
1 1 1 X X X
1 1 1 X X X
Digite a condicao para passar para o proximo estado
1010
Permaneço no estado : [1]
```

This screenshot shows a similar setup to the first one, but with a different terminal output. The file manager window "FSM.pgm - Visualizador de Imagens [1/3]" still shows the same state transition diagram. The terminal window "Terminal - guest-vGwalP@EngSeg-LSEC: ~/Downloads/backup.4.15-12-2" shows a different sequence of state transitions and a table of results.

```
0 1 0 0 0 1
0 1 1 1 0 1
1 0 0 0 0 0
1 0 1 1 0 0
1 1 0 X X X
1 1 1 X X X
Digite a condicao para passar para o proximo estado
1
Passando de: Estado[1] -> Estado[2]
1
Passando de: Estado[2] -> Estado[3]
^C
guest-vGwalP@EngSeg-LSEC:~/Downloads/backup.4.15-12-2017/Tabe
la_Verdade$ ./exe
0
4
1 1
0 0 X 0 0 X
0 0 1 0 1 0
0 1 0 0 0 1
0 1 1 1 0 1
1 0 0 0 0 0
1 0 1 1 0 0
1 1 0 X X X
1 1 1 X X X
Digite a condicao para passar para o proximo estado
1
Passando de: Estado[1] -> Estado[2]
```





EM MOORE:

